



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/706,360	11/12/2003	Barbara Liskov	CIS03-17(7429)	6805
22468	7590	04/18/2007	EXAMINER	
CHAPIN & HUANG L.L.C. WESTBOROUGH OFFICE PARK 1700 WEST PARK DRIVE WESTBOROUGH, MA 01581			PHAN, TUANKHANH D	
			ART UNIT	PAPER NUMBER
			2109	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	04/18/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)
	10/706,360	LISKOV ET AL.
	Examiner TuanKhanh Phan	Art Unit 2109

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 12 November 2003.
- 2a) This action is FINAL. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-27 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-27 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on 12 November 2003 is/are: a) accepted or b) objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 - a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____. |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____. | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| | 6) <input type="checkbox"/> Other: _____. |

DETAILED ACTION

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

2. Claims 1-27 are rejected under 35 U.S.C. 102(e) as being anticipated by Burbeck et al. (US Patent No. 7,143,139).

Regarding claim 1, Burbeck et al. disclose a method for maintaining a client session in a network having a plurality of routers (abstract "persistent broadcast tiers"), the network having an application executed at a plurality of replicas (Col. 28 "one or more nodes") comprising the steps of: providing a database of bindings of request identifiers to replicas where each binding is a record having a request identifier (abstract; Col. 28), a replica identifier and a binding expiration time (Col. 3, lines 9-26 "time interval"; Col. 20, lines 50-59), the

database associated with a first router of the plurality of routers (Col. 3, lines 1-4; abstract; Col. 28, lines 1-37 "first node"); maintaining a change log of records entered into the database, each change log entry having a change event generated by the first router and an event number sequential to an event number of a preceding change event in the change log (Col. 28; Col. 30, lines 5-64 "stored reputation information"); maintaining a current version vector associated with the database and the change log, the current version vector entry for the first router being a most recent event number from the change log, the current version vector entry for each other router being a most recent event number received at the first router from that other router; receiving an update of change events generated at another router in the plurality (Col. 28; Col. 30, lines 5-64 "refreshed information"); reconciling the current version vector according to the received update (Col. 28; Col. 23, lines 30-65; Col. 30, lines 5-64 "refreshed information"); and reconciling the database according to the received update such that the client session is maintained (abstract).

Regarding claim 2, Burbeck et al. disclose the method of claim 1 wherein the request identifier is a client identifier and an application identifier (abstract; Col. 28, lines 1-42).

Regarding claim 3, Burbeck et al. disclose the method of claim 2 wherein the client identifier is an Internet Protocol address (Col. 1, lines 38-56).

Regarding claim 4, Burbeck et al. disclose the method of claim 2 wherein the client identifier is a dproxy Internet Protocol address such that the binding associates a dproxy with a replica (Figure 17A; Col. 1, lines 38-56).

Regarding claim 5, Burbeck et al. disclose the method of claim 1 wherein the step of reconciling the current version vector comprises the steps of: comparing a least recent event number of the router that generated the update to the event number in the current version vector entry, for that router (Col. 23, lines 30-65); if the least recent event number is in series with the event numbers in the database as determined by the current version vector entry for that other router, then entering the most recent event number of the received update into the current version vector ("node's version") entry for the router that generated the update of change events (Col. 23, lines 30-65); and if the least recent event number in the update is not in succession to the event number in the current version vector entry for the router that generated the update of change events, then discarding the received update (Col. 23, lines 30-65).

Regarding claim 6, Burbeck et al. disclose the method of claim 5 wherein the step of reconciling the database further comprises: if the update was not discarded in the step of reconciling the current version vector, then for each entry of the received update, a) determining whether the received entry has expired (Col. 3, lines 5-52); b) if the received entry has expired, then discarding the entry (Col. 3, lines 5-52); c) if the received entry has not expired, then comparing the request identifier of the received entry with the request identifier in the entries in the database (Col. 3, lines 5-52); d) if a matching entry is not found for the received entry, adding the received entry to the database (Col. 3, lines 5-52); e) if a matching entry is found for the received entry, then comparing the application identifier of the received entry with the application identifier of the matching entry

(Col. 3, lines 5-52; Col. 28); f) if the application identifiers match, then retaining the entry having a later expiration time in the database (Col. 3, lines 5-52, Col. 30); and g) if the application identifiers do not match, then retaining an entry selected based on a deterministic function applied to a portion of each entry (abstract; Col. 3, lines 5-52; Col. 28)

Regarding claim 7, Burbeck et al. disclose the method of claim 6 wherein the step of retaining an entry based on a deterministic function comprises the steps of applying a function to the application identifiers (Col. 23; Col. 28); and selecting an entry based on the outcome of the function (Col. 27, lines 62-67; Col. 28, lines 1-42).

Regarding claim 8, Burbeck et al. disclose the method of claim 6 wherein the step of retaining an entry based on a deterministic function comprises the steps of applying the deterministic function to the request identifier (abstract; Col. 27, lines 62-67; Col. 28, lines 1-42) and selecting an entry based on the outcome of the deterministic function (Col. 27, lines 62-67; Col. 28, lines 1-42).

Regarding claim 9, Burbeck et al. disclose the method of claim 1 further comprising the step of deleting a binding from the database when the expiration time for the binding has been exceeded (Col. 23, lines 30-62).

Regarding claim 10, Burbeck et al. disclose the method of claim 1 further comprising the step of sending a request for an update of change events to another router in the plurality (Col. 27, lines 62-67; Col. 28, lines 1-42); and the step of receiving the update further comprises receiving the update in response to the request (Col. 27, lines 62-67; Col. 28, lines 1-42).

Regarding claim 11, Burbeck et al. disclose the method of claim 1 further comprising the steps of: periodically generating a first router update of change events (Col. 27, lines 62-67; Col. 28, lines 1-42); and, transmitting the first router update of change events to at least one other router in the plurality (Col. 27, lines 62-67; Col. 28, lines 1-42).

Regarding claim 12, Burbeck et al. disclose the method of claim 1 further comprising the steps of: affirming that an update has been received from each router of the plurality within a predetermined period for each router (Col. 3, lines 5-48; Col. 27, lines 62-67; Col. 28, lines 1-42); if an update has not been received from a router within the predetermined period for that router, requesting an update of change events from that router (Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67); and if an update is received in response to the request, reconciling the current version vector according to the received update (Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67); and reconciling the database according to the received update (Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67).

Regarding claim 13, Burbeck disclose the method of claim 5 wherein the step of reconciling the database further comprises the steps of: determining from the received update whether the database has a complete record of changes based on the current version vector (Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67); if the database does not have a complete record of changes, requesting a replacement database from a router of the plurality of routers (Col. 23, lines 10-62; Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67).

Regarding claim 14, Burbeck et al. disclose the method of claim 1 further comprising the step of transmitting a copy of the database and the current version vector to another router of the plurality of routers in response to a request from the other router (abstract; Col. 23, lines 1-67; Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67).

Regarding claim 15, Burbeck et al. disclose the method of claim 1 wherein the computerized device fails temporarily and recovers, the method further comprising the steps of: writing the first router change log to a persistent storage device (abstract; Col. 23; Col. 31, lines 63-67; Col. 32, 1-67); sending an update of change events written to the change log in the persistent storage device to other routers in the network (Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67); after recovering from failure, requesting a database and an associated version vector from one of the routers in the plurality (abstract; Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67); retaining the received database and associated version vector (abstract); reconciling the received database with the change log from the persistent storage device (Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67); and updating the received version vector (Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67).

Regarding claim 16, Burbeck et al. disclose the method of claim 1 wherein the received update includes a version vector and the method of maintaining a current version vector further comprises the step of maintaining the current version vector in a version vector table including past version vectors (abstract; Col. 23; Col. 27, lines 62-67; Col. 28, lines 1-42; Col. 30, lines 10-67).

Regarding claim 17, Burbeck et al. disclose the method of claim 16 further comprising the steps of: determining from the version vector table whether the database is current based on the version vector table (abstract; Col. 15, lines 15-67); and if the database is not current, then requesting missed change events from a second router in the network (abstract; Col. 15, lines 15-67).

Regarding claim 18, Burbeck et al. disclose the method of claim 17 wherein each router caches updates received from other routers in the plurality, the method further comprising the step of (abstract; Col. 27, lines 62-67; Col. 28, lines 1-42): if the router that generated the received update does not respond to the request for missed change events, requesting the missed change events from a second router of the plurality and reconciling the change events into the database and current version vector (abstract; Col. 27, lines 62-67; Col. 28, lines 1-42).

Regarding claim 19, Burbeck et al. disclose the method of claim 1 wherein the computerized device fails temporarily and recovers, wherein the step of maintaining a current version vector further comprises the steps of: creating an epoch timestamp from a clock of the computerized device to mark a recovery period (Col. 18; Col. 20, lines 13-67; Col. 21, lines 1-41; Col. 27, lines 62-67; Col. 28, lines 1-42); entering a value pair to the current version vector for the first router, the value pair being an event number and the epoch timestamp (Col. 23, lines 30-63); and the method further comprising the step of after recovery, requesting a database copy and associated version vector from one of the other routers in the plurality (abstract; Col. 17, lines 1- 67; Col 18, lines 1-67).

Regarding claim 20, Burbeck et al. disclose the method of claim 19 further comprising the steps of: determining whether a pre-selected time period has passed (Col. 18; Col. 20, lines 50-63); and deleting value pairs before a most recent value pair from the current version vector having timestamps created before the pre-selected time period (Col. 18; Col. 23, lines 30-63).

Regarding claim 21, Burbeck et al. disclose a system to maintain a client session in a network having a plurality of routers, the network having an application executing at a plurality of replicas, comprising: a network interface to receive a request from a client to access the application (abstract; Col. 27, lines 62-67; Col. 28, lines 1-42); a storage device to store a database of bindings of requests to replicas where each binding is a record of a request identifier (Col. 14; Col. 28), a replica identifier and a binding expiration time (Col. 3), the storage device to further store a change log and a current version vector associated with the change log where the change log includes records added to the database by the system and the current version vector has an entry for each router in the network (Col. 23), each entry storing a version vector of a particular router (Col. 14, lines 5-67; Col. 15, lines 32-63; Col. 19, lines 39-48); and a controller coupled to the interface and the storage device (Col. 14; Col. 28), the controller configured to route the request (Col. 28), if the controller finds a binding matching the request, the controller to route the request to a replica of the plurality according to the binding and to reset the binding expiration time (Col. 18), if the controller does not find a binding matching the request, the controller to add a new record to the database having information from the request to form a binding

Art Unit: 2109

of the client to a replica of the plurality of replicas (abstract; Col. 14, lines 5-67; Col. 15, lines 32-63; Col. 19, lines 39-48; Col. 23, lines 10-67; Col. 27, lines 62-67; Col. 28; lines 1-67).

Regarding claim 22, Burbeck et al. disclose the system of claim 21 wherein the network interface is configured to receive an update of change events for the database from another router in the plurality of routers in the network; and the controller is further configured to reconcile the database according to the received update and to update the current version vector in response to reconciling the database (abstract; Col. 27, lines 62-67; Col. 28, lines 1-42).

Regarding claim 23, Burbeck et al. disclose the system of claim 21 wherein the controller is configured to transmit periodically, to at least one of the other routers in the plurality, an update of change events and the current version vector (abstract; Col. 20, lines 32-67; Col. 27, lines 62-67; Col. 28, lines 1-67).

Regarding claim 24, Burbeck et al. disclose a computerized device to maintain a client session in a network having a plurality of such computerized devices, the network having an application executing at a plurality of replicas, comprising (abstract): means for providing a database of bindings of request identifiers to replicas where each binding is a record having a request identifier, a replica identifier and a binding expiration time, the database associated with a first router of the plurality of routers (abstract; Col. 14, lines 5-67; Col. 15, lines 32-63; Col. 23, lines 10-67; Col. 27, lines 62-67; Col. 28, lines 1-67); means for maintaining a change log of records entered into the database, each change log

entry having a change event generated by the first router and an event number sequential to an event number of a preceding change event in the change log (Col. 28; Col. 30, lines 5-64); means for maintaining a current version vector associated with the database and the change log, the current version vector entry for the first router being a most recent event number from the change log, the current version vector entry for each other router being a most recent event number received at the first router from that other router; means for receiving an update of change events generated at another router in the plurality (abstract; Col. 3, lines 1-4; Col. 28, lines 1-57; Col. 30, lines 5-64); means for reconciling the current version vector according to the received update (Col. 27, lines 62-67; Col. 28, lines 1-42); and means for reconciling the database according to the received update such that the client session is maintained (abstract; Col. 23, lines 10-67; Col. 27, lines 62-67; Col. 28, lines 1-67).

Regarding claim 25, Burbeck et al. disclose a computer program product having a computer-readable medium including computer program logic encoded thereon that, when performed on a computer system having a coupling of a memory, a processor, and at least one communications interface, provides a method for maintaining a client session in a network having an application executing at a plurality of replicas by performing the operations of: providing a database of bindings of request identifiers to replicas where each binding is a record having a request identifier, a replica identifier and a binding expiration time, the database associated with a first router of the plurality of routers (abstract; Col. 14, lines 5-67; Col. 15, lines 32-63; Col. 23, lines 10-67; Col. 27,

lines 62-67; Col. 28, lines 1-67); maintaining a change log of records entered into the database, each change log entry having a change event generated by the first router and an event number sequential to an event number of a preceding change event in the change log (Col. 28; Col. 30, lines 5-64); maintaining a current version vector associated with the database and the change log, the current version vector entry for the first router being a most recent event number from the change log, the current version vector entry for each other router being a most recent event number received at the first router from that other router (Col. 3; Col. 23, lines 5-55; Col. 28; Col. 30, lines 5-64); receiving an update of change events generated at another router in the plurality (Col. 23, lines 5-55; Col. 28; Col. 30, lines 5-64); reconciling the current version vector according to the received update (Col. 3, lines 5-55; Col. 17, lines 4-62; Col. 28; Col. 30, lines 5-64); and reconciling the database according to the received update such that the client session is maintained (abstract; Col. 28; Col. 30, lines 5-64).

Regarding claim 26, Burbeck et al. disclose a method in a computerized device for maintaining a client session in a network having a plurality of routers, the network having an application executed at a plurality of replicas (Col. 28), comprising the steps of: providing a database of bindings of requests from clients to replicas where each binding is a record having a request from a client, a replica identifier and a binding expiration time, the database associated with a first router of the plurality of routers (abstract; Col. 14, lines 5-67; Col. 15, lines 32-63; Col. 23, lines 10-67; Col. 27, lines 62-67; Col. 28, lines 1-67); maintaining a change log of records entered into the database, each change log entry having

a change event generated by the first router and an event number sequential to an event number of a preceding change event in the change log (Col. 14, lines 5-67; Col. 15, lines 32-63; Col. 19, lines 39-48); maintaining a current version vector associated with the database and the change log, the current version vector entry for the first router being a most recent event number from the change log, the current version vector entry for each other router being a most recent event number received at the first router from that other router (abstract; Col. 14, lines 5-67; Col. 15, lines 32-63; Col. 23, lines 10-67; Col. 27, lines 62-67; Col. 28, lines 1-67); receiving a request from the client to access the application, the request having a client identifier and an application identifier (abstract; Col. 14, lines 5-67; Col. 15, lines 32-63; Col. 19, lines 39-48); comparing data from the request with records in the database (Col. 19; Col. 20); if the database has a record with a client identifier and an application identifier matching the data from the request, a) routing the request to a replica of the plurality of replicas according to the matching record (Col. 19; Col. 20); and b) resetting the binding expiration time of the matching record (Col. 20); and if the database does not have a record with a client identifier and a replica identifier matching the data from the request, a) routing the request to one of the plurality of replicas (Col. 19; Col. 20); and b) entering, into the database a new record forming a binding of the request to one of the replicas (abstract; Col. 3; Col. 27, lines 62-67; Col. 28, lines 1-67).

Regarding claim 27, Burbeck et al. disclose the method of claim 26 wherein the router is a DNS server and wherein the application identifier in the

request is a domain name and wherein the step of routing comprises mapping the request to an Internet Protocol address of the one replica (abstract; Col. 3; 27, lines 62-67; Col. 28, lines 1-67).

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1-4, 10, 12, 14, 16, and 21-27 are further rejected under 35 U.S.C. 103(a) as being unpatentable over Srivastava (US Patent 7,047,315) in view of Rajsbaum (*ACM SIGACT News Distributed Computing Column 8*).

Regarding claims 1, 21, and 24-26, Srivastava teaches methods, systems, and devices for maintaining a client session in a network having a plurality of routers (abstract), the network, network interfaces, storage and performance having applications executed at a plurality of replicas (abstract; Col. 30, lines 53-67; Col. 31, lines 1-44; Col. 32, lines 11-67; Col. 33, lines 1-60), comprising the steps of: providing a database of bindings of request identifiers to replicas where each binding is a record having a request identifier (abstract), a replica identifier and a binding expiration time (Col. 22, lines 35-62), the database associated with a first router of the plurality of routers (abstract; Col. 30, lines 53-67; Col. 31, lines 1-44; Col. 32, lines 11-67; Col. 33, lines 1-60); maintaining a change log of

records entered into the database (abstract; Col. 30, lines 53-67; Col. 31, lines 1-44; Col. 32, lines 11-67; Col. 33, lines 1-60), receiving an update of change events generated at another router in the plurality (Col. 31); reconciling the current version vector according to the received update (Col. 31); and reconciling the database according to the received update such that the client session is maintained (abstract; Col. 30, lines 53-67; Col. 31, lines 1-44; Col. 32, lines 11-67; Col. 33, lines 1-60).

Srivastava does not appear to explicitly teach the event-based or a change event generation.

However, from the same field of endeavor of maintaining a distributed network system, Rajsbaum teaches a distribution system having an each change log entry having a change event generated by the first router and an event number sequential to an event number of a preceding change event in the change log (p. 57 "event based").

At the time of the invention, it would have been obvious to one having ordinary skill in the art, having the teachings of Srivastava and Rajsbaum before him or her, to combine the maintaining of a client session of Srivastava to include the "event-based" teaching of Rajsbaum because it would reduce network latency and bandwidth.

Regarding claim 2, Srivastava teaches the request identifier is a client identifier and an application identifier (abstract; Col. 30, lines 53-67; Col. 31, lines 1-44).

Regarding claim 3, Srivastava teaches the client identifier is an Internet Protocol address (Col. 1, lines 40-61).

Regarding claim 4, Srivastava teaches the client identifier is a dproxy Internet Protocol address such that the binding associates a dproxy with a replica (Col. 1, lines 40-61).

Regarding claim 10, Srivastava teaches the step of sending a request for an update of change path list to another router in the plurality (Col. 31, lines 1-44); and the step of receiving the update further comprises receiving the update in response to the request (abstract; Col. 30, lines 53-67; Col. 31, lines 1-44; Col. 33, lines 1-60).

Regarding claim 12, Srivastava teaches the steps of: affirming that an update has been received from each router of the plurality within a predetermined period for each router (abstract); if an update has not been received from a router within the predetermined period for that router, requesting an update of change events from that router (Col. 30, lines 53-67; Col. 31, lines 1-44); and if an update is received in response to the request, reconciling the current version vector according to the received update (Col. 30, lines 53-67; Col. 31); and reconciling the database according to the received update (Col. 31).

Regarding claims 14 and 16, Srivastava teaches the step of transmitting and maintaining a copy of the database and the current version vector to another router of the plurality of routers in response to a request from the other router request (abstract; Col. 30, lines 53-67; Col. 31, lines 1-44; Col. 33, lines 1-60).

Claims 22 and 23 are rejected for the same reason as discussed in claim 21 above.

Regarding claim 27, Srivastava teaches the router is a DNS server and wherein the application identifier in the request is a domain name and wherein the step of routing comprises mapping the request to an Internet Protocol address of the one replica (Col. 1, lines 25-67).

Conclusion

5. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure:

Garcia-Luna-Aceves et al. Pub. No. 20030101278. May 29, 2003 (*System and Method for Directing Clients to Optimal Servers in Computer Networks*).

Liskov, Barbara. US Patent 7,158,985. Jan 2, 2007. *Method and Apparatus for Efficient Propagation of Large Datasets under Failure Conditions*.

Oracle9i Advanced Replication Release 2 (9.2). *Master Replication Concepts and Architecture*. December 5, 2002.

Radoslavov, Pavlin et al. *Topology-Informed Internet Replica Placement*. 2001.

Saito, Yasyshi et al. *Taming Aggressive Replication in the Pangaea Wide-Area File System*. USENIX Association. Winter 2002. pp. 15-30.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to TuanKhanh Phan whose telephone number is 571-270-3047. The examiner can normally be reached on Mon to Fri, 9:00am to 5:00pm EST, 1st Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Patrick Assouad can be reached on 571-272-2210. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

tkp



PATRICK ASSOUAD
SUPERVISORY PATENT EXAMINER